In the Claims:

Please cancel claims 3, 4, 12 and 16-19 without prejudice or dedication.

Please amend the claims as indicated below.

1. (currently amended) A method for monitoring system processor usage time of~~by~~ a software agent operating in a computer system, said method comprising the steps of:

identifying said agent by associating an agent identifier therewith;

initiating, responsive to said identifying of said agent, an agent lifetime timer for measuring~~monitoring~~ an operating interval of~~for~~ said agent;

determining said operating interval using said lifetime timer by identifying a start time and a completion time of said agent and computing an elapsed time as the difference between said starting time and said completion time for said agent; and

storing said operating interval and said agent identifier in a computer-readable memory.

2. (original) The method of claim 1, wherein said computer-readable memory includes a hash table.

3. (canceled)

4. (canceled)

5. (original) The method of claim 1 further comprising:

associating said operating interval and said agent identifier with other operating intervals and agent identifiers associated with a plurality of other software agents operating in said system.

6. (original) The method of claim 5 further comprising:

filtering said agent and said plurality of other agents according to predefined filtering criteria to produce a filtered set.

7. (original) The method of claim 6 further comprising:

rank ordering said filtered set.

8. (original) The method of claim 7 further comprising:

making said filtered set available to a display device.

9. (original) The method of claim 6 further comprising:

determining a corrective measure for at least one member of said filtered set.

10. (original) The method of claim 9 further comprising:

displaying said corrective measure on a display device.

11. (original) The method of claim 9, wherein said corrective measure is implemented by said system.

12. (canceled)

13. (currently amended) A method for monitoring system processor time usage of~~by~~ a software agent created by~~having~~ a thread associated therewith, said thread having a thread lifetime and said agent having an agent lifetime, said method comprising the steps of:

associating an agent identifier with said agent;

initiating, responsive to said associating said agent identifier with said agent, an agent lifetime timer for monitoring said agent lifetime;

determining system processor resource allocations of~~associated with~~ said agent, by identifying a start time and a completion time of said agent and computing said agent lifetime as the difference between said starting time and said completion time for said agent, said resource allocations defining a footprint for said agent comprising:

an amount of system processor resources utilized by said thread during said thread lifetime; and

an amount of system processor resources utilized by said agent during said agent lifetime;

associating said footprint with said agent identifier;

storing said footprint and said agent identifier in a computer-readable memory;

comparing said footprint of said agent to a plurality of footprints associated with a like plurality of other software agents;

ranking said footprint of said agent against said plurality of footprints; and

displaying those of said agent footprint and said plurality of footprints exceeding a predefined threshold.

14. (previously presented) The method of claim 13 further comprising:

establishing a system processor resources configuration threshold defining a maximum amount of system processor resources to be utilized by each of said software agent and said plurality of other software agents.

15. (previously presented) The method of claim 13, further comprising:

running a collection probe to determine if a total amount of consumed system processor resources exceeds said configuration threshold; and

performing said initiating step when said total amount of consumed system processor resources exceeds said configuration threshold.

16. (canceled)

17. (canceled)

18. (canceled)

19. (canceled)

20. (currently amended) A method for tracking system processor time of~~for~~ a target agent operatively associated with a hypertext transport protocol process operating on a computer system and running a plurality of threads, said target agent further creating~~operating with~~ at least one of said plurality of threads, said method comprising:

creating a computer-readable hash table in a memory operatively associated with said computer system;

initiating an agent tracking function in machine-executable code in said computer system;

identifying members of said plurality of threads by associating a thread identifier with each member of said plurality of threads producing a like plurality of identified threads;

identifying those of said plurality of identified threads created by~~having~~ said target agent to produce ~~operating therewith producing~~ an identified thread set;

determining an amount of said system processor time utilized by said identified thread set; and

storing said system processor time for said identified thread set in said hash table, thereby tracking said system processor time of~~for~~ said target agent.

21. (previously presented) The method of claim 20 further comprising:

computing statistics for said identified thread set.

22. (previously presented) The method of claim 20 further comprising:

rank ordering those of said plurality of identified threads having said target agent operating therewith.

23. (previously presented) The method of claim 22 further comprising:

providing said identified set to a display device.